COMPUTER-AIDED IMAGE ANALYSIS

RELATED APPLICATIONS

This application claims the benefit of priority of U.S. provisional application Serial No. 60/263,381 filed January 23, 2001. This application is also a continuation-in-part of application Serial No. 09/633,410, filed August 7, 2000, which is a continuation-in-part of application Serial No. 09/578,011, filed May 24, 2000, which is a continuation-in-part of application Serial No. 09/568,301, filed May 9, 2000, now issued as Patent No. _____________, which is a continuation of application Serial No. 09/303,387. filed May 1, 1999, now issued as Patent No. 6,128,608, which claims priority to U.S. provisional application Serial No. 60/083,961, filed May 1, 1998. This application is related to co-pending applications Serial No. 09/633,615, Serial No. 09/633,616, and Serial No. 09/633,850, all filed August 7, 2000, which are also continuations-in-part of application Serial No. 09/578,011. This application is also related to applications Serial No. 09/303,386 and Serial No. 09/305,345, now issued as Patent No. 6,157,921, both filed May 1, 1999, and to application Serial No. 09/715,832, filed November 14, 2000, all of which also claim priority to provisional application Serial No. 60/083,961.

FIELD OF THE INVENTION

The present invention relates generally to computer-aided analysis of images and more particularly to computer-aided image analysis using support vector machines.

BACKGROUND OF THE INVENTION

Optimal extraction of data contained within an electromagnetic signal requires the ability to identify important components of the signal in spite of noise and limitations of the signal source and the instrumentation used to detect the signal. A key area in which optimized extraction and reconstruction of data is sought is the field of image analysis, where sources of noise and other factors can negatively impact the ability to efficiently extract data from the image, thus impairing the effectiveness of the imaging method for its intended use. Examples of areas in which image analysis can be problematic include astronomical observation and planetary exploration, where sources can be faint and atmospheric interference introduce noise and distortion, military and security

surveillance, where light can be low and rapid movement of targets result in low contrast and blur, and medical imaging, which often suffers from low contrast, blur and distortion due to source and instrument limitations. Adding to the difficulty of image analysis is the large volume of data contained within a digitized image, since the value of any given data point often cannot be established until the entire image is processed.

Development of methods for automated analysis of digital images has received considerable attention over that past few decades, with one of the key areas of interest being the medical field. Applications include analysis of pathology images generated using visual, ultrasound, x-ray, positron emission, magnetic resonance and other imaging methods. As in the case of human-interpreted medical images, an automated image analyzer must be capable of recognizing and classifying blurred features within the images, which often requires discrimination of faint boundaries between areas differing by only a few gray levels or shades of color.

In recent years, machine-learning approaches for image analysis have been widely explored for recognizing patterns which, in turn, allow extraction of significant features within an image from a background of irrelevant detail. Learning machines comprise algorithms that may be trained to generalize using data with known outcomes. Trained learning machine algorithms may then be applied to predict the outcome in cases of unknown outcome. Machine-learning approaches, which include neural networks, hidden Markov models, belief networks and support vector machines, are ideally suited for domains characterized by the existence of large amounts of data, noisy patterns and the absence of general theories. Particular focus among such approaches has been on the application of artificial neural networks to biomedical image analysis, with results reported in the use of neural networks for analyzing visual images of cytology specimens and mammograms for the diagnosis of breast cancer, classification of retinal images of diabetics, karyotyping (visual analysis of chromosome images) for identifying genetic abnormalities, and tumor detection in ultrasound images, among others.

The majority of learning machines that have been applied to image analysis are neural networks trained using back-propagation, a gradient-based method in which errors in classification of training data are propagated backwards through the network to adjust the bias weights of the network elements until the mean squared error is minimized. A significant drawback of back-propagation neural networks is that the empirical risk

function may have many local minimums, a case that can easily obscure the optimal solution from discovery. Standard optimization procedures employed by back-propagation neural networks may converge to a minimum, but the neural network method cannot guarantee that even a localized minimum is attained, much less the desired global minimum. The quality of the solution obtained from a neural network depends on many factors. In particular, the skill of the practitioner implementing the neural network determines the ultimate benefit, but even factors as seemingly benign as the random selection of initial weights can lead to poor results. Furthermore, the convergence of the gradient-based method used in neural network learning is inherently slow. A further drawback is that the sigmoid function has a scaling factor, which affects the quality of approximation. Possibly the largest limiting factor of neural networks as related to knowledge discovery is the "curse of dimensionality" associated with the disproportionate growth in required computational time and power for each additional feature or dimension in the training data.

The shortcomings of neural networks can be overcome by using another type of learning machine -- the support vector machine. In general terms, a support vector machine maps input vectors into high dimensional feature space through a non-linear mapping function, chosen *a priori*. In this high dimensional feature space, an optimal separating hyperplane is constructed. The optimal hyperplane is then used to determine perform operations such as class separations, regression fit, or density estimation.

Within a support vector machine, the dimensionally of the feature space may be very high. For example, a fourth degree polynomial mapping function causes a 200 dimensional input space to be mapped into a 1.6 billion dimensional feature space. The kernel trick and the Vapnik-Chervonenkis ("VC") dimension allow the support vector machine to avoid the "curse of dimensionality" that typically limits other methods and effectively derive generalizable answers from this very high dimensional feature space.

If the training vectors are separated by the optimal hyperplane (or generalized optimal hyperplane), the expected value of the probability of committing an error on a test example is bounded by the examples in the training set. This bound depends on neither the dimensionality of the feature space, the norm of the vector of coefficients, nor the bound of the number of the input vectors. Therefore, if the optimal hyperplane can be constructed from a small number of support vectors relative to the training set size, the

generalization ability will be high, even in infinite dimensional space.

As such, support vector machines provide a desirable solution for the problem of analyzing a digital image from vast amounts of input data. However, the ability of a support vector machine to analyze a digitized image from a data set is limited in proportion to the information included within the training data set. Accordingly, there exists a need for a system and method for pre-processing data so as to augment the training data to maximize the computer analysis of an image by the support vector machine.

BRIEF SUMMARY OF THE INVENTION

The system and method for analyzing digitized images uses a learning machine in general and a support vector machine in particular. A training data set consisting of digital image data generated from imaging a biological or medical subject with known outcome is pre-processed to allow the most advantageous application of the learning machine. For purposes of the present invention, the image can be derived *ex vivo* , e.g., a tissue sample viewed through a microscope, or *in vivo*, e.g., an x-ray projection image. Each training data point comprises a vector having one or more coordinates. Pre-processing the training data set comprises identifying missing or erroneous data points and taking appropriate steps to correct the flawed data or, as appropriate, remove the observation or the entire field from the scope of the problem. Pre-processing the training data set may also comprise adding dimensionality to each training data point by adding one or more new coordinates to the vector. The new coordinates added to the vector may be derived by applying a transformation to one or more of the original coordinates. The transformation may be based on expert knowledge, or may be computationally derived. In a situation where the training data set comprises a continuous variable, the transformation may comprise optimally categorizing the continuous variable of the training data set.

The support vector machine is trained using the pre-processed training data set. In this manner, the additional representations of the training data provided by the preprocessing enhances the learning machine's ability to analyze the data therefrom. In the particular context of support vector machines, the greater the dimensionality of the training set, the higher the quality of the generalizations that may be derived therefrom.

When the analysis to be performed from the data relates to a regression or density estimation or where the training output comprises a continuous variable, the training output may be post-processed by optimally categorizing the training output to derive categorizations from the continuous variable.

A test data set is pre-processed in the same manner as was the training data set. Then, the trained learning machine is tested using the pre-processed test data set. A test output of the trained learning machine may be post-processed to determine if the test output is an optimal solution. Post-processing the test output may comprise interpreting the test output into a format that may be compared with the test data set. Alternative post-processing steps may enhance the human interpretability or suitability for additional processing of the output data.

In the context of a support vector machine, a method is provided for the selection of a kernel prior to training the support vector machine. The selection of a kernel may be based on prior knowledge of the specific problem being addressed or analysis of the properties of any available data to be used with the learning machine and is typically dependant on the nature of the analysis to be made from the data. Optionally, an iterative process comparing post-processed training outputs or test outputs can be applied to make a determination as to which configuration provides the optimal solution. If the test output is not the optimal solution, the selection of the kernel may be adjusted and the support vector machine may be retrained and retested. When it is determined that the optimal solution has been identified, a live data set, i.e., a data set with unknown results, may be collected and pre-processed in the same manner as was the training data set. The pre-processed live data set is input into the learning machine for processing. The live output of the learning machine may then be post-processed by interpreting the live output into a computationally derived alphanumeric classifier.

In an exemplary embodiment, a system is provided for analysis of a digitized image from image data using a support vector machine. The exemplary system comprises a storage device for storing a database containing a training data set and a test data set, each data set comprising image data, and a processor for executing one or more support vector machines. The processor is also operable for collecting the training data set from the database, pre-processing the training data set to enhance each of a plurality of training data points, training the support vector machine using the pre-processed

training data set, collecting the test data set from the database, pre-processing the test data set in the same manner as was the training data set, testing the trained support vector machine using the pre-processed test data set, and in response to receiving the test output of the trained support vector machine, post-processing the test output to determine if the test output is an optimal solution. The exemplary system may also comprise a communications device for receiving the test data set and the training data set from a remote source. In such a case, the processor may be operable to store the training data set in the storage device prior to pre-processing of the training data set and to store the test data set in the storage device prior to pre-processing of the test data set. The exemplary system may also comprise a display device for displaying the post-processed test data. The processor of the exemplary system may further be operable for performing each additional function described above. The communications device may be further operable to send a computationally-derived alphanumeric classifier to a remote source.

In an exemplary image analysis sequence using kernel-based learning machines, in particular, support vector machines, digitized image data are input into the processor where a detection component identifies the areas (objects) of particular interest in the image and, by segmentation, separates those objects from the background. A feature extraction component formulates numerical values relevant to the classification task from the segmented objects. Results of the preceding analysis steps are input into a support vector machine classifier which produces an output which may consist of an index discriminating between two possible diagnoses, or some other output in the desired output format. Additional support vector machines may be included to assist in the segmentation or feature extraction components prior.

In a preferred embodiment, digitized image data are input into a plurality of subsystems, each subsystem having one or more kernel-based learning machine. Each subsystem analyzes the data relevant to a different feature or characteristic found within the image. For example, using the example of mammogram analysis, one subsystem may look at and classify calcifications, another subsystem may look at and classify masses, while a third subsystem looks at and classifies structural distortions. Once each subsystem completes its analysis and classification, the output for all subsystems is input into an overall kernel-based, e.g., support vector machine, analyzer which combines the data to make a diagnosis, decision or other action which utilizes the knowledge obtained

from the image.

Specific procedures for the preprocessing of data and training of support vector machines is described in U.S. Patent Nos. 6,157,921 and 6,128,608 which are incorporated herein by reference in their entirety. For processing of image data, pre-processing may include the use of known transformations which facilitate extraction of the useful data. Such transformations may include, but are not limited to, Fourier transforms, wavelet transforms, Radon transforms and Hough transforms.

BRIEF DESCRIPTION OF THE DRAWINGS

Exemplary embodiments of the present invention will hereinafter be described with reference to the below-listed drawings, in which like numerals indicate like elements throughout the figures.

FIG. 1 is a flowchart illustrating an exemplary general method for analyzing data using a learning machine.

FIG. 2 is a flowchart illustrating an exemplary method for analyzing data using a support vector machine.

FIG. 3 is a flowchart illustrating an exemplary optimal categorization method that may be used in a stand-alone configuration or in conjunction with a learning machine for pre-processing or post-processing techniques.

FIG. 4 illustrates an exemplary unexpanded data set that may be input into a support vector machine.

FIGs. 5a and 5b are diagrams of gray scale features in an image, where FIG. 5a illustrates the un-processed image and FIG. 5b illustrates the image after segmentation pre-processing.

FIG. 6 illustrates an exemplary expanded data set that may be input into a support vector machine.

FIG. 7 illustrates an exemplary input data set for a standalone application of the optimal categorization method of FIG. 3.

FIG. 8 is a functional block diagram illustrating an exemplary operating environment for an exemplary embodiment of the present invention.

FIG. 9 is a functional block diagram illustrating a hierarchical system of multiple support vector machines.

FIG. 10 is a functional block diagram illustrating a basic process flow for image analysis using support vector machines.

FIG. 11 is a functional block diagram illustrating an exemplary image analysis system with multiple detection subsystems for use in analysis of mammograms.

FIG. 12 is a combined curve and bit mapped image illustrating mapping of gray levels to a gray level curve.

FIG. 13 is a bit mapped image following feature extraction processing of calcification images containing in a mammogram.

FIG. 14 is a diagram illustrating a pre-processing transformation for converting image segments to fixed dimensional form.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The following detailed description utilizes a number of acronyms which are generally well known in the art. While definitions are typically provided with the first instance of each acronym, for convenience, Table 1 below provides a list of the acronyms and abbreviations used herein along with their respective definitions.

ACRONYM	DESCRIPTION
ATAPI	attachment packet interface
CT	computed tomography
DMA	direct memory access
EIDE	enhanced integrated drive electronics
FFT	fast Fourier transform
I/O	input/output
IDE	integrated drive electronics
LAN	local area network
MRI	magnetic resonance imagining
PET	positron emission tomography
RAM	random access memory
ROM	read-only memory
SCSI	small computer system interface
SPECT	single-photon emission computed tomography
SVM	support vector machine
WAN	wide area network

Table 1

The present invention provides improved methods for analyzing images using learning machines. As used herein, the term “image” means the product of any imaging method, whether the image is obtained through conventional visual methods, e.g., photography, or by any other method of detecting an electromagnetic signal impinging on a recording medium or device, e.g., infrared radiation impinging on an infrared detector.

Of particular interest in the described examples are the medical imaging methods, including but not limited to, x-ray, PET (positron emission tomography), MRI (magnetic resonance imaging), CT (computed tomography), SPECT (single-photon emission computed tomography), gamma camera, confocal microscopy (also referred to as "visual"), electrical impedance imaging, and ultrasound. For purposes of the present invention, the image can be derived *ex vivo* , e.g., a tissue sample viewed through a microscope, or *in vivo*, e.g., an x-ray projection image. For imaging methods that generate analog outputs, the analog output will have been digitized, either by digital scanning or by converting an analog signal into a digital signal such that input image to be analyzed according to the present invention is presumed to be in digital form.

While several examples of learning machines exist and advancements are expected in this field, the exemplary embodiments of the present invention focus on the support vector machine.

A first aspect of the present invention facilitates image analysis by optionally pre-processing the data prior to using the data to train a learning machine and/or optionally post-processing the output from a learning machine. Generally stated, pre-processing data comprises reformatting or augmenting the data in order to allow the learning machine to be applied most advantageously. For example, evaluation of one or more important characteristics within an image may involve pre-processing to create a bit map from the original gray scale image, or features of varying sizes may need to be converted, i.e., normalized, to a fixed dimensional form prior to processing in order to permit comparison of qualities such as contour, shape or density.

In a manner similar to pre-processing, post-processing involves interpreting the output of a learning machine in order to discover meaningful characteristics thereof. The meaningful characteristics to be ascertained from the output may be problem- or data-specific. Post-processing involves interpreting the output into a form that, for example, may be understood by or is otherwise useful to a human observer, or converting the output into a form which may be readily received by another device for, e.g., archival or transmission.

FIG. 1 is a flowchart illustrating a general method 100 for analyzing data using learning machines. The method 100 begins at starting block 101 and progresses to step 102 where a specific problem is formalized for application of analysis through machine

learning. Particularly important is a proper formulation of the desired output of the learning machine. For instance, in predicting future performance of an individual equity instrument, or a market index, a learning machine is likely to achieve better performance when predicting the expected future change rather than predicting the future price level. The future price expectation can later be derived in a post-processing step as will be discussed later in this specification.

After problem formalization, step 103 addresses training data collection. Training data comprises a set of data points having known characteristics. Training data may be collected from one or more local and/or remote sources. The collection of training data may be accomplished manually or by way of an automated process, such as known electronic data transfer methods. Accordingly, an exemplary embodiment of the learning machine for use in conjunction with the present invention may be implemented in a networked computer environment. Exemplary operating environments for implementing various embodiments of the learning machine will be described in detail with respect to FIGS. 10-11.

At step 104, the collected training data is optionally pre-processed in order to allow the learning machine to be applied most advantageously toward extraction of the knowledge inherent to the training data. During this preprocessing stage the training data can optionally be expanded through transformations, combinations or manipulation of individual or multiple measures within the records of the training data. As used herein, "expanding data" is meant to refer to altering the dimensionality of the input data by changing the number of observations available to determine each input point (alternatively, this could be described as adding or deleting columns within a database table). By way of illustration, a data point may comprise the coordinates (1,4,9). An expanded version of this data point may result in the coordinates (1,1,4,2,9,3). In this example, it may be seen that the coordinates added to the expanded data point are based on a square-root transformation of the original coordinates. By adding dimensionality to the data point, this expanded data point provides a varied representation of the input data that is potentially more meaningful for analysis by a learning machine. Data expansion in this sense affords opportunities for learning machines to analyze data not readily apparent in the unexpanded training data.

Expanding data may comprise applying any type of meaningful transformation to

the data and adding those transformations to the original data. The criteria for determining whether a transformation is meaningful may depend on the input data itself and/or the type of knowledge that is sought from the data. Illustrative types of data transformations include: addition of expert information; labeling; binary conversion, e.g., a bit map; transformations, such as Fourier, wavelet, Radon, principal component analysis and kernel principal component analysis, as well as clustering; scaling; normalizing; probabilistic and statistical analysis; significance testing; strength testing; searching for two-dimensional regularities; Hidden Markov Modeling; identification of equivalence relations; application of contingency tables; application of graph theory principles; creation of vector maps; addition, subtraction, multiplication, division, application of polynomial equations and other algebraic transformations; identification of proportionality; determination of discriminatory power; etc. In the context of medical data, potentially meaningful transformations include: association with known standard medical reference ranges; physiologic truncation; physiologic combinations; biochemical combinations; application of heuristic rules; diagnostic criteria determinations; clinical weighting systems; diagnostic transformations; clinical transformations; application of expert knowledge; labeling techniques; application of other domain knowledge; Bayesian network knowledge; etc. Specifically with regard to medical imaging, transformations can include segmentation techniques to recognize homogeneous regions within an image as distinct and belonging to different objects. Image segmentation techniques include histogram thresholding, edge-based segmentation, tree/graph based approaches, region growing, mass contraction, clustering, probabilistic or Bayesian approaches, neural networks for segmentation, and others. These and other transformations, as well as combinations thereof, will occur to those of ordinary skill in the art.

Those skilled in the art should also recognize that data transformations may be performed without adding dimensionality to the data points. For example a data point may comprise the coordinate (A, B, C). A transformed version of this data point may result in the coordinates (1, 2, 3), where the coordinate "1" has some known relationship with the coordinate "A," the coordinate "2" has some known relationship with the coordinate "B," and the coordinate "3" has some known relationship with the coordinate "C." A transformation from letters to numbers may be required, for example, if letters are not understood by a learning machine. Other types of transformations are possible

without adding dimensionality to the data points, even with respect to data that is originally in numeric form. Furthermore, it should be appreciated that pre-processing data to add meaning thereto may involve analyzing incomplete, corrupted or otherwise "dirty" data. A learning machine cannot process "dirty" data in a meaningful manner. Thus, a pre-processing step may involve cleaning up or filtering a data set in order to remove, repair or replace dirty data points.

Returning to FIG. 1, the exemplary method 100 continues at step 106, where the learning machine is trained using the pre-processed data. As is known in the art, a learning machine is trained by adjusting its operating parameters until a desirable training output is achieved. The determination of whether a training output is desirable may be accomplished either manually or automatically by comparing the training output to the known characteristics of the training data. A learning machine is considered to be trained when its training output is within a predetermined error threshold from the known characteristics of the training data. In certain situations, it may be desirable, if not necessary, to post-process the training output of the learning machine at step 107. As mentioned, post-processing the output of a learning machine involves interpreting the output into a meaningful form. In the context of a regression problem, for example, it may be necessary to determine range categorizations for the output of a learning machine in order to determine if the input data points were correctly categorized. In the example of a pattern recognition problem, it is often not necessary to post-process the training output of a learning machine.

At step 108, test data is optionally collected in preparation for testing the trained learning machine. Test data may be collected from one or more local and/or remote sources. In practice, test data and training data may be collected from the same source(s) at the same time. Thus, test data and training data sets can be divided out of a common data set and stored in a local storage medium for use as different input data sets for a learning machine. Regardless of how the test data is collected, any test data used must be pre-processed at step 110 in the same manner as was the training data. As should be apparent to those skilled in the art, a proper test of the learning may only be accomplished by using testing data of the same format as the training data. Then, at step 112 the learning machine is tested using the pre-processed test data, if any. The test output of the learning machine is optionally post-processed at step 114 in order to determine if the

results are desirable. Again, the post processing step involves interpreting the test output into a meaningful form. The meaningful form may be one that is readily understood by a human or one that is compatible with another processor. Regardless, the test output must be post-processed into a form which may be compared to the test data to determine whether the results were desirable. Examples of post-processing steps include but are not limited of the following: optimal categorization determinations, scaling techniques (linear and non-linear), transformations (linear and non-linear), and probability estimations. The method 100 ends at step 116.

FIG. 2 is a flow chart illustrating an exemplary method 200 for enhancing knowledge that may be discovered from data using a specific type of learning machine known as a support vector machine (SVM). A SVM implements a specialized algorithm for providing generalization when estimating a multi-dimensional function from a limited collection of data. A SVM may be particularly useful in solving dependency estimation problems. More specifically, a SVM may be used accurately in estimating indicator functions (e.g. pattern recognition problems) and real-valued functions (e.g. function approximation problems, regression estimation problems, density estimation problems, and solving inverse problems). The SVM was originally developed by Vladimir N. Vapnik. The concepts underlying the SVM are explained in detail in his book, entitled *Statistical Leaning Theory* (John Wiley & Sons, Inc. 1998), which is herein incorporated by reference in its entirety. Accordingly, a familiarity with SVMs and the terminology used therewith are presumed throughout this specification.

The exemplary method 200 begins at starting block 201 and advances to step 202, where a problem is formulated and then to step 203, where a training data set is collected. As was described with reference to FIG. 1, training data may be collected from one or more local and/or remote sources, through a manual or automated process. At step 204 the training data is optionally pre-processed. Again, pre-processing data comprises enhancing meaning within the training data by cleaning the data, transforming the data and/or expanding the data. Those skilled in the art should appreciate that SVMs are capable of processing input data having extremely large dimensionality. In fact, the larger the dimensionality of the input data, the better the generalizations a SVM is able to calculate. Therefore, while training data transformations are possible that do not expand the training data, in the specific context of SVMs it is preferable that training data be

expanded by adding meaningful information thereto.

At step 206 a kernel is selected for the SVM. As is known in the art, different kernels will cause a SVM to produce varying degrees of quality in the output for a given set of input data. Therefore, the selection of an appropriate kernel may be essential to the desired quality of the output of the SVM. In one embodiment of the learning machine, a kernel may be chosen based on prior performance knowledge. As is known in the art, exemplary kernels include polynomial kernels, radial basis classifier kernels, linear kernels, etc. In an alternate embodiment, a customized kernel may be created that is specific to a particular problem or type of data set. In yet another embodiment, the multiple SVMs may be trained and tested simultaneously, each using a different kernel. The quality of the outputs for each simultaneously trained and tested SVM may be compared using a variety of selectable or weighted metrics (see step 222) to determine the most desirable kernel. In a preferred embodiment for image processing, a Fourier kernel is selected to address issues of geometric shape recognition. This Fourier kernel, described in more detail below, is invariant under transformations of translation and rotation.

Next, at step 208 the pre-processed training data is input into the SVM. At step 210, the SVM is trained using the pre-processed training data to generate an optimal hyperplane. Optionally, the training output of the SVM may then be post-processed at step 211. Again, post-processing of training output may be desirable, or even necessary, at this point in order to properly calculate ranges or categories for the output. At step 212 test data is collected similarly to previous descriptions of data collection. The test data is pre-processed at step 214 in the same manner as was the training data above. Then, at step 216 the pre-processed test data is input into the SVM for processing in order to determine whether the SVM was trained in a desirable manner. The test output is received from the SVM at step 218 and is optionally post-processed at step 220.

Based on the post-processed test output, it is determined at step 222 whether an optimal minimum was achieved by the SVM. Those skilled in the art should appreciate that a SVM is operable to ascertain an output having a global minimum error. However, as mentioned above, output results of a SVM for a given data set will typically vary with kernel selection. Therefore, there are in fact multiple global minimums that may be ascertained by a SVM for a given set of data. As used herein, the term "optimal

minimum" or "optimal solution" refers to a selected global minimum that is considered to be optimal (e.g. the optimal solution for a given set of problem specific, pre-established criteria) when compared to other global minimums ascertained by a SVM. Accordingly, at step 222, determining whether the optimal minimum has been ascertained may involve comparing the output of a SVM with a historical or predetermined value. Such a predetermined value may be dependant on the test data set. For example, in the context of a pattern recognition problem where data points are classified by a SVM as either having a certain characteristic or not having the characteristic, a global minimum error of 50% would not be optimal. In this example, a global minimum of 50% is no better than the result that would be achieved by flipping a coin to determine whether the data point had that characteristic. As another example, in the case where multiple SVMs are trained and tested simultaneously with varying kernels, the outputs for each SVM may be compared with output of other SVM to determine the practical optimal solution for that particular set of kernels. The determination of whether an optimal solution has been ascertained may be performed manually or through an automated comparison process.

If it is determined that the optimal minimum has not been achieved by the trained SVM, the method advances to step 224, where the kernel selection is adjusted. Adjustment of the kernel selection may comprise selecting one or more new kernels or adjusting kernel parameters. Furthermore, in the case where multiple SVMs were trained and tested simultaneously, selected kernels may be replaced or modified while other kernels may be re-used for control purposes. After the kernel selection is adjusted, the method 200 is repeated from step 208, where the pre-processed training data is input into the SVM for training purposes. When it is determined at step 222 that the optimal minimum has been achieved, the method advances to step 226, where live data is collected similarly as described above. By definition, live data has not been previously evaluated, so that the desired output characteristics that were known with respect to the training data and the test data are not known.

At step 228 the live data is pre-processed in the same manner as was the training data and the test data. At step 230, the live pre-processed data is input into the SVM for processing. The live output of the SVM is received at step 232 and is post-processed at step 234. In one embodiment of the learning machine, post-processing comprises converting the output of the SVM into a computationally-derived alpha-numerical

classifier for interpretation by a human or computer. Preferably, the alphanumerical classifier comprises a single value that is easily comprehended by the human or computer. The method 200 ends at step 236.

FIG. 3 is a flow chart illustrating an exemplary optimal categorization method 300 that may be used for pre-processing data or post-processing output from a learning machine. Additionally, as will be described below, the exemplary optimal categorization method may be used as a stand-alone categorization technique, independent from learning machines. The exemplary optimal categorization method 300 begins at starting block 301 and progresses to step 302, where an input data set is received. The input data set comprises a sequence of data samples from a continuous variable. The data samples fall within two or more classification categories. Next, at step 304 the bin and class-tracking variables are initialized. As is known in the art, bin variables relate to resolution, while class-tracking variables relate to the number of classifications within the data set. Determining the values for initialization of the bin and class-tracking variables may be performed manually or through an automated process, such as a computer program for analyzing the input data set. At step 306, the data entropy for each bin is calculated. Entropy is a mathematical quantity that measures the uncertainty of a random distribution. In the exemplary method 300, entropy is used to gauge the gradations of the input variable so that maximum classification capability is achieved.

The method 300 produces a series of "cuts" on the continuous variable, such that the continuous variable may be divided into discrete categories. The cuts selected by the exemplary method 300 are optimal in the sense that the average entropy of each resulting discrete category is minimized. At step 308, a determination is made as to whether all cuts have been placed within input data set comprising the continuous variable. If all cuts have not been placed, sequential bin combinations are tested for cutoff determination at step 310. From step 310, the exemplary method 300 loops back through step 306 and returns to step 308 where it is again determined whether all cuts have been placed within input data set comprising the continuous variable. When all cuts have been placed, the entropy for the entire system is evaluated at step 309 and compared to previous results from testing more or fewer cuts. If it cannot be concluded that a minimum entropy state has been determined, then other possible cut selections must be evaluated and the method proceeds to step 311. From step 311 a heretofore untested selection for number of cuts is

chosen and the above process is repeated from step 304. When either the limits of the resolution determined by the bin width has been tested or the convergence to a minimum solution has been identified, the optimal classification criteria is output at step 312 and the exemplary optimal categorization method 300 ends at step 314.

The optimal categorization method 300 takes advantage of dynamic programming techniques. As is known in the art, dynamic programming techniques may be used to significantly improve the efficiency of solving certain complex problems through carefully structuring an algorithm to reduce redundant calculations. In the optimal categorization problem, the straightforward approach of exhaustively searching through all possible cuts in the continuous variable data would result in an algorithm of exponential complexity and would render the problem intractable for even moderate sized inputs. By taking advantage of the additive property of the target function, in this problem the average entropy, the problem may be divide into a series of sub-problems. By properly formulating algorithmic sub-structures for solving each sub-problem and storing the solutions of the sub-problems, a significant amount of redundant computation may be identified and avoided. As a result of using the dynamic programming approach, the exemplary optimal categorization method 300 may be implemented as an algorithm having a polynomial complexity, which may be used to solve large sized problems.

As mentioned above, the exemplary optimal categorization method 300 may be used in pre-processing data and/or post-processing the output of a learning machine. For example, as a pre-processing transformation step, the exemplary optimal categorization method 300 may be used to extract classification information from raw data. As a post-processing technique, the exemplary optimal range categorization method may be used to determine the optimal cut-off values for markers objectively based on data, rather than relying on ad hoc approaches. As should be apparent, the exemplary optimal categorization method 300 has applications in pattern recognition, classification, regression problems, etc. The exemplary optimal categorization method 300 may also be used as a stand-alone categorization technique, independent from SVMs and other learning machines. An exemplary stand-alone application of the optimal categorization method 300 will be described with reference to FIG. 7.

In an example of pre-processing of data use in image analysis, image segmentation provides means for isolating objects from the background to emphasize the

salient features of the original image. Quite often, particularly in medical applications, two or more objects may be overlapped or clustered together. For example, in two-dimensional gel image analysis, several spots can cluster together. In cell imaging, cells can overlap. In mammograms, calcifications and masses can overlap. In such cases, separation of the objects is crucial in an effective analysis system.

Referring to FIG. 5a, two partially overlapping masses 502, 504 represented as a gray scale image are illustrated. In an exemplary embodiment, a "gravitation" model is iteratively applied to the gray scale image to contract the masses. In the digital image, pixel values are viewed as "mass" values, and gravitational forces among the masses are used for the contraction movements. The process is analogous to the process of star and planet formation. The initially wide spread masses 502, 504 are contracted under the gravitation model toward the respective centroids to produce two dense, well-formed bodies shown in FIG. 5b as 502′ and 504′. This approach is driven by the natural patterns in the image itself. No prior information about the specifics of the image is required. The gravitation model is insensitive to noise and outliers, and is generic in that it is applicable to different types of images by simply adjusting the threshold for pixel movements. In general principle, the gravitation model might be considered an inverse of region growing algorithms which are known in image segmentation, however, instead of expanding from a "seed", the object contracts into a "seed" so that distinct seeds can be identified. Alternatively, other known image segmentation algorithms may be used to pre-process the image data to enhance the image analysis process.

FIG. 4 illustrates an exemplary unexpanded data set 400 that may be used as input for a support vector machine. This data set 400 is referred to as "unexpanded" because no additional information has been added thereto. As shown, the unexpanded data set comprises a training data set 402 and a test data set 404. Both the unexpanded training data set 402 and the unexpanded test data set 404 comprise data points, such as exemplary data point 406, relating to historical clinical data from sampled medical patients. In this example, the data set 400 may be used to train a SVM to determine whether a breast cancer patient will experience a recurrence or not.

Each data point includes five input coordinates, or dimensions, and an output classification shown as 406a-f which represent medical data collected for each patient. In particular, the first coordinate 406a represents "Age," the second coordinate 406b

represents "Estrogen Receptor Level," the third coordinate 406c represents "Progesterone Receptor Level," the fourth coordinate 406d represents "Total Lymph Nodes Extracted," the fifth coordinate 406e represents "Positive (Cancerous) Lymph Nodes Extracted," and the output classification 406f, represents the "Recurrence Classification." The important known characteristic of the data 400 is the output classification 406f (Recurrence Classification), which, in this example, indicates whether the sampled medical patient responded to treatment favorably without recurrence of cancer ("-1") or responded to treatment negatively with recurrence of cancer ("1"). This known characteristic will be used for learning while processing the training data in the SVM will be used in an evaluative fashion after the test data is input into the SVM thus creating a "blind" test, and will obviously be unknown in the live data of current medical patients.

Table 2 provides an exemplary test output from a SVM trained with the unexpanded training data set 402 and tested with the unexpanded data set 404 shown in FIG. 4.

Vapnik's Polynomial	
Alphas bounded up to 1000	
Input values will be individually scaled to lie between 0 and 1	
SV zero threshold: 1e-16	
Margin threshold: 0.1	
Objective zero tolerance: 1e-17	
Degree of polynomial: 2	
Test set:	
Total samples:	24
Positive samples:	8
False negatives:	4
Negative samples:	16
False positives:	6

Table 2

The test output has been post-processed to be comprehensible by a human or computer. According to the table, the test output shows that 24 total samples (data points) were examined by the SVM and that the SVM incorrectly identified four of eight positive samples (50%), i.e., found negative for a positive sample, and incorrectly identified 6 of sixteen negative samples (37.5%), i.e., found positive for a negative sample.

FIG. 6 illustrates an exemplary expanded data set 600 that may be used as input for a support vector machine. This data set 600 is referred to as "expanded" because

additional information has been added thereto. Note that aside from the added information, the expanded data set 600 is identical to the unexpanded data set 400 shown in FIG. 4. The additional information supplied to the expanded data set has been supplied using the exemplary optimal range categorization method 300 described with reference to FIG. 3. As shown, the expanded data set comprises a training data set 602 and a test data set 604. Both the expanded training data set 602 and the expanded test data set 604 comprise data points, such as exemplary data point 606, relating to historical data from sampled medical patients. Again, the data set 600 may be used to train a SVM to learn whether a breast cancer patient will experience a recurrence of the disease.

Through application of the exemplary optimal categorization method 300, each expanded data point includes twenty coordinates (or dimensions) 606a1-3 through 606e1-3, and an output classification 606f, which collectively represent medical data and categorization transformations thereof for each patient. In particular, the first coordinate 606a represents "Age," the second coordinate through the fourth coordinate 606a1 - 606a3 are variables that combine to represent a category of age. For example, a range of ages may be categorized, for example, into "young" "middle-aged" and "old" categories respective to the range of ages present in the data. As shown, a string of variables "0" (606a1), "0" (606a2), "1" (606a3) may be used to indicate that a certain age value is categorized as "old." Similarly, a string of variables "0" (606a1), "1" (606a2), "0" (606a3) may be used to indicate that a certain age value is categorized as "middle-aged." Also, a string of variables "1" (606a1), "0" (606a2), "0" (606a1) may be used to indicate that a certain age value is categorized as "young." From an inspection of FIG. 6, it may be seen that the optimal categorization of the range of "Age" 606a values, using the exemplary method 300, was determined to be 31-33 = "young," 34 = "middle-aged" and 35-49 = "old." The other coordinates, namely coordinate 606b "Estrogen Receptors Level," coordinate 606c "Progesterone Receptor Level," coordinate 606d "Total Lymph Nodes Extracted," and coordinate 606e "Positive (Cancerous) Lymph Nodes Extracted," have each been optimally categorized in a similar manner.

Table 3 provides an exemplary expanded test output from a SVM trained with the expanded training data set 602 and tested with the expanded data set 604 shown in FIG. 6.

Vapnik's Polynomial	
Alphas bounded up to 1000	
Input values will be individually scaled to lie between 0 and 1	
SV zero threshold: 1e-16	
Margin threshold: 0.1	
Objective zero tolerance: 1e-17	
Degree of polynomial: 2	
Test set:	
Total samples:	24
Positive samples:	8
False negatives:	4
Negative samples:	16
False positives:	4

Table 3

The expanded test output has been post-processed to be comprehensible by a human or computer. As indicated, the expanded test output shows that 24 total samples (data points) were examined by the SVM and that the SVM incorrectly identified four of eight positive samples (50%) and incorrectly identified four of sixteen negative samples (25%). Accordingly, by comparing this expanded test output with the unexpanded test output of Table 2, it may be seen that the expansion of the data points leads to improved results (i.e. a lower global minimum error), specifically a reduced instance of patients who would unnecessarily be subjected to follow-up cancer treatments.

FIG. 7 illustrates an exemplary input and output for a stand alone application of the optimal categorization method 300 described in FIG. 3. In the example of FIG. 8, the input data set 801 comprises a "Number of Positive Lymph Nodes" 802 and a corresponding "Recurrence Classification" 804. In this example, the optimal categorization method 300 has been applied to the input data set 801 in order to locate the optimal cutoff point for determination of treatment for cancer recurrence, based solely upon the number of positive lymph nodes collected in a post-surgical tissue sample. The well-known clinical standard is to prescribe treatment for any patient with at least three positive nodes. However, the optimal categorization method 300 demonstrates that the optimal cutoff , seen in Table 4, based upon the input data 801, should be at the higher value of 5.5 lymph nodes, which corresponds to a clinical rule prescribing follow-up treatments in patients with at least six positive lymph nodes.

Number of subintervals: 2 Number of classes: 2 Number of data points: 46 Lower bound: -1 Upper bound: 10 Number of bins: 22 Regularization constant: 1 Data file: posnodes.prn Min. Entropy - 0.568342 Optimal cut-off: 5.500000

Table 4

As shown in Table 5 below, the prior art accepted clinical cutoff point (≥ 3.0) resulted in 47% correctly classified recurrences and 71% correctly classified non-recurrences.

Cut Point	Correctly Classified Recurrence	Correctly Classified Non-Recurrence
Clinical (≥3.0)	7 of 15 (47%)	22 of 31 (71%)
Optimal (≥5.5))	5 of 15 (33%)	30 of 31 (97%)

Table 5

Accordingly, 53% of the recurrences were incorrectly classified (further treatment was improperly not recommended) and 29% of the non-recurrences were incorrectly classified (further treatment was incorrectly recommended). By contrast, the cutoff point determined by the optimal categorization method 300 (≥ 5.5) resulted in 33% correctly classified recurrences and 97% correctly classified non-recurrences. Accordingly, 67% of the recurrences were incorrectly classified (further treatment was improperly not recommended) and 3% of the non-recurrences were incorrectly classified (further treatment was incorrectly recommended).

As shown by this example, it may be feasible to attain a higher instance of correctly identifying those patients who can avoid the post-surgical cancer treatment regimes, using the exemplary optimal categorization method 300. Even though the cutoff point determined by the optimal categorization method 300 yielded a moderately higher percentage of incorrectly classified recurrences, it yielded a significantly lower percentage of incorrectly classified non-recurrences. Thus, considering the trade-off, and realizing that the goal of the optimization problem was the avoidance of unnecessary

treatment, the results of the cutoff point determined by the optimal categorization method 300 are mathematically superior to those of the prior art clinical cutoff point. This type of information is potentially extremely useful in providing additional insight to patients weighing the choice between undergoing treatments such as chemotherapy or risking a recurrence of breast cancer.

Table 6 is a comparison of exemplary post-processed output from a first support vector machine comprising a linear kernel and a second support vector machine comprising a polynomial kernel.

I. Simple Dot Product	**II. Vapnik's Polynomial**
Alphas bounded up to 1000. Input values will not be scaled. SV zero threshold: 1e-16 Margin threshold: 0.1 Objective zero tolerance: 1e-07	Alphas bounded up to 1000. Input values will not be scaled. SV zero threshold: 1e-16 Margin threshold: 0.1 Objective zero tolerance: 1e-07 Degree of polynomial: 2
<u>Test set</u> Total samples: 24 Positive samples: 8 False negatives: 6 Negative samples: 16 False positives: 3	<u>Test set</u> Total samples: 24 Positive samples: 8 False negatives: 2 Negative samples: 16 False positives: 4

Table 6

Table 6 demonstrates that a variation in the selection of a kernel may affect the level of quality of the output of a SVM. As shown, the post-processed output of a first SVM (Column I) comprising a linear dot product kernel indicates that for a given test set of twenty four samples, six of eight positive samples were incorrectly identified and three of sixteen negative samples were incorrectly identified. By way of comparison, the post-processed output for a second SVM (Column II) comprising a polynomial kernel indicates that for the same test set, only two of eight positive samples were incorrectly identified and four of sixteen negative samples were identified. By way of comparison, the polynomial kernel yielded significantly improved results pertaining to the identification of positive samples and yielded only slightly worse results pertaining to the identification of negative samples. Thus, as will be apparent to those of skill in the art, the global minimum error for the polynomial kernel is lower than the global minimum error for the linear kernel for this data set.

FIG. 8 and the following discussion are intended to provide a brief and general description of a suitable computing environment for implementing the computer-aided

image analysis of the present invention. Although the system shown in FIG. 8 is a conventional personal computer 1000, those skilled in the art will recognize that the invention also may be implemented using other types of computer system configurations. The computer 1000 includes a central processing unit 1022, a system memory 1020, and an Input/Output ("I/O") bus 1026. A system bus 1021 couples the central processing unit 1022 to the system memory 1020. A bus controller 1023 controls the flow of data on the I/O bus 1026 and between the central processing unit 1022 and a variety of internal and external I/O devices. The I/O devices connected to the I/O bus 1026 may have direct access to the system memory 1020 using a Direct Memory Access ("DMA") controller 1024.

The I/O devices are connected to the I/O bus 1026 via a set of device interfaces. The device interfaces may include both hardware components and software components. For instance, a hard disk drive 1030 and a floppy disk drive 1032 for reading or writing removable media 1050 may be connected to the I/O bus 1026 through disk drive controllers 1040. An optical disk drive 1034 for reading or writing optical media 1052 may be connected to the I/O bus 1026 using a Small Computer System Interface ("SCSI") 1041. Alternatively, an IDE (Integrated Drive Electronics, i.e., a hard disk drive interface for PCs), ATAPI (ATtAchment Packet Interface, i.e., CD-ROM and tape drive interface), or EIDE (Enhanced IDE) interface may be associated with an optical drive such as may be the case with a CD-ROM drive. The drives and their associated computer-readable media provide nonvolatile storage for the computer 1000. In addition to the computer-readable media described above, other types of computer-readable media may also be used, such as ZIP drives, or the like.

A display device 1053, such as a monitor, is connected to the I/O bus 1026 via another interface, such as a video adapter 1042. A parallel interface 1043 connects synchronous peripheral devices, such as a laser printer 1056, to the I/O bus 1026. A serial interface 1044 connects communication devices to the I/O bus 1026. A user may enter commands and information into the computer 1000 via the serial interface 1044 or by using an input device, such as a keyboard 1038, a mouse 1036 or a modem 1057. Other peripheral devices (not shown) may also be connected to the computer 1000, such as audio input/output devices or image capture devices.

A number of program modules may be stored on the drives and in the system

memory 1020. The system memory 1020 can include both Random Access Memory ("RAM") and Read Only Memory ("ROM"). The program modules control how the computer 1000 functions and interacts with the user, with I/O devices or with other computers. Program modules include routines, operating systems 1065, application programs, data structures, and other software or firmware components. In an illustrative embodiment, the learning machine may comprise one or more pre-processing program modules 1075A, one or more post-processing program modules 1075B, and/or one or more optimal categorization program modules 1077 and one or more SVM program modules 1070 stored on the drives or in the system memory 1020 of the computer 1000. Specifically, pre-processing program modules 1075A, post-processing program modules 1075B, together with the SVM program modules 1070 may comprise computer-executable instructions for pre-processing data and post-processing output from a learning machine and implementing the learning algorithm according to the exemplary methods described with reference to FIGS. 1 and 2. Furthermore, optimal categorization program modules 1077 may comprise computer-executable instructions for optimally categorizing a data set according to the exemplary methods described with reference to FIG. 3.

The computer 1000 may operate in a networked environment using logical connections to one or more remote computers, such as remote computer 1060. The remote computer 1060 may be a server, a router, a peer device or other common network node, and typically includes many or all of the elements described in connection with the computer 1000. In a networked environment, program modules and data may be stored on the remote computer 1060. The logical connections depicted in FIG. 8 include a local area network ("LAN") 1054 and a wide area network ("WAN") 1055. In a LAN environment, a network interface 1045, such as an Ethernet adapter card, can be used to connect the computer 1000 to the remote computer 1060. In a WAN environment, the computer 1000 may use a telecommunications device, such as a modem 1057, to establish a connection. It will be appreciated that the network connections shown are illustrative and other devices of establishing a communications link between the computers may be used.

In another embodiment, a plurality of SVMs can be configured to hierarchically process multiple data sets in parallel or sequentially. In particular, one or more first-level

SVMs may be trained and tested to process a first type of data and one or more first-level SVMs can be trained and tested to process a second type of data. Additional types of data may be processed by other first-level SVMs. The output from some or all of the first-level SVMs may be combined in a logical manner to produce an input data set for one or more second-level SVMs. In a similar fashion, output from a plurality of second-level SVMs may be combined in a logical manner to produce input data for one or more third-level SVM. The hierarchy of SVMs may be expanded to any number of levels as may be appropriate. In this manner, lower hierarchical level SVMs may be used to pre-process data that is to be input into higher level SVMs. Also, higher hierarchical level SVMs may be used to post-process data that is output from lower hierarchical level SVMs.

Each SVM in the hierarchy or each hierarchical level of SVMs may be configured with a distinct kernel. For example, SVMs used to process a first type of data may be configured with a first type of kernel while SVMs used to process a second type of data may utilize a second, different type of kernel. In addition, multiple SVMs in the same or different hierarchical level may be configured to process the same type of data using distinct kernels.

FIG. 9 is presented to illustrate an exemplary hierarchical system of SVMs. As shown, one or more first-level SVMs 1302a and 1302b may be trained and tested to process a first type of input data 1304a, such as mammography data, pertaining to a sample of medical patients. One or more of these SVMs may comprise a distinct kernel, indicated as "KERNEL 1" and "KERNEL 2". Also, one or more additional first-level SVMs 1302c and 1302d may be trained and tested to process a second type of data 1304b, which may be, for example, genomic data or images of cytology specimens, for the same or a different sample of medical patients. Again, one or more of the additional SVMs may comprise a distinct kernel, indicated as "KERNEL 1" and "KERNEL 3". The output from each of the like first-level SVMs may be compared with each other, e.g., 1306a compared with 1306b; 1306c compared with 1306d, in order to determine optimal outputs 1308a and 1308b. Then, the optimal outputs from the two groups or first-level SVMs, i.e., outputs 1308a and 1308b, may be combined to form a new multi-dimensional input data set 1310, for example, relating to mammography and genomic data. The new data set may then be processed by one or more appropriately trained and tested second-level SVMs 1312a and 1312b. The resulting outputs 1314a and 1314b from second-level

SVMs 1312a and 1312b may be compared to determine an optimal output 1316. Optimal output 1316 may identify causal relationships between the mammography and genomic data points. As should be apparent to those of skill in the art, other combinations of hierarchical SVMs may be used to process either in parallel or serially, data of different types in any field or industry in which analysis of data is desired.

In application to image analysis, multiple SVMs are used to process data of different types that can be extracted from a digitized image. The different types of data can comprise different characteristics or qualities of objects found in the image, for example, size, shape, density, quantity, orientation, etc. The following example provides an illustrative application of multiple SVMs to image analysis, particularly for analysis of mammograms for diagnosis of breast cancer.

Calcification in breast tissue is of concern because of its association, in certain configurations, with carcinoma. Computer-aided detection and classification of microcalcifications identified by mammography has been an important area of focus in the field of image analysis. (See, e.g., Abstracts from IWDM 2000 -- Fifth International Workshop on Digital Mammography.) Since a significant percentage of normal screening mammograms show some calcification, mere detection of all calcification provides little benefit since not all types of calcification have the same clinical significance. Generally speaking, microcalcifications are associated with a malignant process and macrocalcifications are associated with a benign process. However, other characteristics of the calcifications can indicate association with either a benign or malignant structure, including shape, number and distribution. Therefore, the ability to distinguish between benign calcifications and those associated with cancer is key to successful computer-aided image analysis of mammograms.

Two additional categories of suspicious abnormalities that may be seen in mammograms which indicate the possible presence of a malignancy are masses and structural distortions. Masses are three-dimensional lesions which may represent a localizing sign of cancer. Masses are described by their location, size, shape, margin characteristics, x-ray attenuation (radiodensity), and effect on surrounding tissue. Structural distortions are focal disruptions of the normal tissue patterns. Radiographically, distortions appear as surrounding tissue being "pulled inward" into a focal point

FIG. 10 provides a flowchart of the basic analysis sequence according to the present invention for mammogram analysis using SVMs. The digitized mammogram image 1102 is input into the processor where the detection component 1104 finds the areas (objects) of particular interest in the image 1102 and, by segmentation, separates these objects from the background. The feature extraction component 1106 formulates numerical values relevant to the classification task from the segmented objects. The SVM classifier 1108 produces an index discriminating between the benign and malignant cases.

Implementation of the exemplary embodiment of the inventive image analysis system and method for mammogram analysis employs three SVM-based detection subsystems for calcifications 1202, masses 1204 and structural distortions 1206, each of which receives the digitized mammogram images 1201 as input, as shown in FIG. 11. Although each of the three subsystems was developed separately, the basic structure of each subsystem is similar. The outputs of the three subsystems are input into a separate SVM 1250 which performs overall analysis and provides the final output, which in this case, would be a diagnosis indicating the presence or absence of a malignancy.

In each of the three subsystems, the detection component finds the areas of particular interest in the image and separates the objects from the background. The feature extraction component formulates numerical values relevant to the classification task from the segmented objects. The SVM classifier produces an index discriminating between the benign and malignant cases.

The individual components can be developed in parallel due to their modular structure. (See, e.g., module 1070 in FIG. 8.) For example, in developing the calcification segmentation component 1202, a selected set of malignant, benign, and normal cases representing a wide range of images was used to guide and test the design in order to produce a general, robust and accurate algorithm. At the same time, the SVM classifier 1242 was developed and tested with manually prepared input data. A set of 300 images (150 benign and 150 malignant cases) was used in training the SVM. An independent set of 328 images was used for testing. High dimensional input features were used to ensure a sufficient capacity for automatically extracted features. The components will be integrated and adjusted for optimal performance.

In calcification detection subsystem 1202, the first step in finding calcifications is

to process the image data to find the bright spots on the mammogram, i.e., to segment the calcifications (step 1212). In the preferred embodiment, the method involves finding local extremes of 2-dimensional discrete function F (x, y). Given that the mammogram consists of gray scale images, the problem involves distinguishing between the white and black spots in the image. The conventional method of solving this problem is to determine for each point (x, y), e.g., each pixel, that the value F(x, y) in any one point is not less then the value in every neighbor point. Images in the computer have eight neighbors for every point (pixel). Another existing method for identifying local minima and maxima involves applying a Gaussian filter to every point (x, y) where the function F(x, y) is determined. Other methods of solving the problem involve finding the local extremes, however, all of the known methods 1) require a number of calculations to be performed at each point, and 2) must be applied to each and every point (pixel) in the image. As a result, these algorithms can be very time consuming.

In one aspect of the present invention, a method for finding local extremes of 2-dimensional discrete function avoids the examination of all points (x, y) and, therefore, dramatically reduces the processing time. Specifically, local maxima and minima are determined by using spots in the image rather than performing a pixel-by-pixel evaluation of brightness. The spots in the image are compared against a series of brightness thresholds to generate a plurality of bitmaps. The method can be illustrated using the case of the gray scale image shown in FIG. 12 as an example. By definition, the brightness of the image $F(x_i, y_j)$ in the computer is a discrete function. Brightness can be further discriminated by decreasing the number of levels of brightness to N (for example, N = 32, or 16, or any other value). The gray image is then transformed into a set of N binary (black ("1") and white ("0")) images (bitmaps). At bitmap L (L = 1, 2, ..., N) the pixel is black if the brightness of the corresponding pixel at the initial image F is greater than F_L, where $F_L = (L-1)\cdot(F_{max} - F_{min})/N$. Otherwise, the pixel is white. Referring to FIG. 12, the dark center of the right-hand image is mapped to the highest level bitmap ("level N") and corresponds to the local maximum. The next lower level bitmap ("level N-1") defines another threshold such that the values on the curve above level N-1 are dark for the N-1 level bitmap. This results in identification of two types of spots -- those that have values above level N and those that have values above level N-1, such that spots with brightness levels exceeding level N will also be included in the level N-1 bitmap.

To differentiate the spots, the two bitmaps (from level N and level N-1) are superimposed. Spots of the first type are spots on level N-1, referred to as "bottom spots." The remaining spots on the level N-1 bitmap represent the "top spots", as indicated in FIG. 12. The bottom spots represent slopes of the curves for the local maxima of the top spots. This process is repeated by superimposing the bitmap from the level N-2 with the bitmap from the level N-1 to identify new top spots and bottom spots at these levels, e.g. the (N-1) top spot and the (N-2) bottom spot. This process is further repeated until all local maxima, i.e. top spots, and bottom spots for each of the N levels are found, thus avoiding the need to perform a pixel-by-pixel analysis of the image.

Calcifications can be classified by describing the geometry of the bright spots. The method of analyzing the geometry of the spots is based on the bitmaps described above for rapid calculation of continuing characteristics. For example, the gradients of slopes corresponding to the spots can be analyzed to distinguish certain background features. It is known that the spots with a low gradient are created by intersection of blood vessels or connecting tissues. On the other hand, spots with very steep slopes are created mainly by artifacts (damages in the emulsion). To estimate the gradient, one uses the border or perimeter of the spot corresponding to the local maximum, i.e., the "upper border", and the border or perimeter of the spot, which represents the slope, i.e., the "bottom border". Because the difference in brightness between the upper and lower borders is known [(Fmax - Fmin)/N], the distance between these borders (in number of pixels, for example) is proportional to the value of the gradient at the slope. Thus, determination of the gradient can be done at a very low computational cost because the binary bitmaps that were already prepared at the previous step for finding bright spots (local maximums) are used, and the only additional requirement is that the number of pixels between the borders be counted. It should be noted that since the spots are often asymmetric and irregular in shape (particularly those associated with a malignancy), this distance may be different in different directions. Therefore, the slope may have different gradients on different directions.

Another aspect of calcification detection subsystem 1202 is to classify the spots as calcifications or non-calcifications. For this purpose, several characteristics of the spot are calculated including, but not limited to: 1) the area of the top spot, 2) the area of the bottom spot, 3) the length of the top border, 4) the length of the bottom border, 5) the

area-to-border ratio for the top spot, 6) the area-to-border ratio for the bottom spot. To separate the calcifications from other bright spots, a pattern recognition technique based on SVM machines is used.

In most problems of image interpretation, the context of each part of an image must be taken into consideration. This is true for the problem of identifying calcifications in mammograms as well. At least three characteristics of the surrounding area of a given bright spot at level L should be considered: 1) the total area of spots at the level L-1 inside a circle of radius RI around the top spot, 2) the proximity of other objects with more prominent characteristics of calcification, and 3) whether the spot is located on a blood vessel. (Vascular calcifications can be seen as parallel tracks or linear tubular calcifications that run along a blood vessel and are typically classified as benign.) As a result of such non-local approach, the following procedure of finding calcifications is used:

A. Find a bright spot.

B. Calculate the geometrical characteristics.

C. Use the SVM to recognize the prominent calcifications.

D. Soften the restrictions for calcification recognition and apply these criteria in the vicinity of the prominent calcifications.

E. Determine whether the "calcification" is located on a vessel and, if so, delete it.

The following provides a method for identifying blood vessels in step E. For this purpose, each spot at each binary bitmap is analyzed as follows:

E_1 Find the border pixels.

E_2 Keep the kernel pixels which are common to opposite borders (left and right borders or top and bottom borders).

E_3 Delete the kernel pixels belonging to the upper border.

E_4 Find the border pixels.

E_5 Delete the border pixels belonging to the right border.

E_6 Find the border pixels.

E_7 Delete the border pixels belonging to the bottom border.

E_8 Find the border pixels.

E_9 Delete the border pixels belonging to the left border.

E_{10} Return to point E_1 and repeat all steps until all pixels on the bitmap are kernel pixels.

The preceding sequence of steps E_1-E_{10} for identification of vessels will transform each spot that is generally shaped as a strip, i.e., elongated as a vessel would be, into what looks like a central line (a set of connected pixels), or a "skeleton" of the strip, as shown in the upper image of FIG. 13. For spots that are not shaped as a strip, i.e., not a vessel, the set of kernel pixels determined according to steps E_1-E_{10} will not create a connected line of appropriate length, thus indicating that the spot is not a vessel. See, e.g., the lower image of FIG. 13.

Clusters of micro-calcifications are characterized by their relatively small sizes and high densities. The algorithm combines a recursive peak seeking technique with morphological operations to achieve a highly accurate calcification detection and segmentation.

Segmentation to distinguish overlapping or closely positioned objects according to the preferred embodiment is described above with reference to FIG. 5, and therefore will not be repeated. Briefly, however, where overlapping calcifications are identified, a gravitation model is applied to contract the objects to allow them to be distinguished.

Following Calcification Segmentation (step 1212), Local SVM analyzer 1222 analyzes the characteristics of individual calcifications detected by the segmentation algorithm. A quantitative measure of the likelihood of a calcification being associated with malignancy is produced by the SVM. All the evaluations from the first stage local SVM analyzer 1222 are used by the second stage SVM 1242 for a more global assessment of the cluster.

For a given SVM, the input data must have the same dimension. Because segmented calcifications will vary in sizes, proper transformations are necessary to convert the variable size image segments to a fixed dimensional form without losing critical information. The following transformation sequence converts the contour of a calcification to a fixed dimensional vector and is illustrated in FIG. 14.

1. Compute the centroid 902 of the calcification 900.

2. Use the centroid 902 as the origin of a polar coordinate system and sample the contour of the calcification with n equally spaced angles. This gives n radial measures 904 which form an n dimensional vector $[r_1, r_2, \ldots, r_n]$.

3. Apply a discrete Fourier transform to the vector obtained in step 2. The resulting n-dimensional complex vector is used as the input to the SVM.

Because n is the predetermined number of sampling radial rays, the dimension of the resulting vector is fixed regardless of input calcification size. This approach avoids the unnatural re-sampling or padding. The Fourier transform takes advantage of the periodic nature of the sampling scheme and further enhances the essential features such as the rotational invariants.

Referring again to FIG. 11, the result of the Local SVM analysis step 1222 is then processed for feature extraction (step 1232). Features known to be relevant in discriminating malignant and benign calcifications are extracted and the results are fed to the Global SVM classifier 1242. Useful features include the number of calcifications, areas, perimeters, locations, orientations, and eccentricities of the calcifications.

Due to the ability of SVMs to process high dimensional input data without sacrificing generalization, a large number of features can be added to the input. Even though the contribution of an individual feature to the classifier may be small, the entire set of features can collectively provide the SVM with sufficient information to achieve proper classification.

An important component in any SVM or other kernel-based method is the kernel used to define the inner product in the feature space. The kernel describes the similarities between the input vectors in a non-linear fashion. The performance of a kernel-based system is largely dependent upon the proper design of the kernel that captures the essential features of the given problem. In the preferred embodiment, a Fourier kernel is used to specifically address the problem of geometric shape recognition and classification. It is clearly desirable that the kernel be invariant under the transformations of translations and rotation. The detected contour from an image will also vary in size. The kernel needs to be robust enough to accommodate a large range of shape patterns while still being sensitive enough to maintain critical information for classification. Given a contour, the Fourier kernel is computed as follows.

1. Given a contour that is a Jordan (simple continuous closed) curve in the plane, represent the contour as a complex-valued function $z(s), 0 \leq s \leq 1$. Regard the origin of the complex plane at the centroid of the contour and associate the points on the contour with the complex numbers of the function.

2. Compute the Fourier coefficients of $z(s)$ up to order N.

$$f_n = \int_0^1 z(s)e^{-2\pi ins}ds, \quad -N \le n \le N \tag{1}$$

3. For two contours $z(s), w(s)$ with Fourier coefficients f_n, g_n, the kernel is defined as

$$K(z,w) = \sum_{n=-N}^{N} | f_n \cdot g_n | \tag{2}$$

The Fourier kernel has many advantages over other kernels in dealing with the shape classification problem in that: 1) the Fourier kernel is translation and rotation invariant. A translated or rotated shape will be considered exactly the same as the original one by the kernel. The invariance is accomplished completely automatically and transparently in the design of the kernel. It does not require any costly alignments or searches. 2) The Fourier kernel is faithful in retaining critical information for shape classification. The Fourier series is an exact representation of the original contour. With a finite number of terms, it is still an accurate approximation to the original. The rotational feature is filtered out in a natural way without affecting other essential features. 3) The Fourier kernel is computationally efficient. A small number of terms (e.g. N=10) is usually sufficient for most practical applications. It can also take advantage of existing fast algorithms such as Fast Fourier Transform (FFT) to achieve greater efficiency.

Other types of transforms which are well known in the art can be used to facilitate extraction of useful data from the original image data rather than analyzing the image data directly. One such transform, the "wavelet transform", provides a powerful tool for multiresolution analysis of the images. Wavelet transforms localize a function both in space and scaling. The coefficients in the wavelet transforms can be used as features at certain scales for the SVM classifier.

Another type of transform, the "Radon transform", maps image points in the space domain to a sinusoidal curve in the Radon transform domain to provide parameters of all possible curves on which the point may lie. An important property of the Radon transform is to extract lines (curves) from very noisy images. Two-dimensional Radon transforms can generate numerical descriptions of many useful features related to the shapes of objects, including convexity, elongation, angularity, and the number of lobes.

(For a discussion of use of the two dimensional Radon transform for analysis of shape, see Leavers, V.F., "Use of the Two-Dimensional Radon Transform to Generate a Taxonomy of Shape for the Characterization of Abrasive Powder Particles", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No.23, 12/2000 which is incorporated herein by reference.) The Hough transform, a special case of the Radon transform, is a standard tool in image analysis that allows recognition of global patterns in an image space by recognition of local patterns (ideally a point) in a transformed parameter space. It is particularly useful when the patterns sought are sparsely digitized, have holes and/or the images are noisy. (The Radon function available in the Image Processing Toolbox of commercially-available MatLab® software (The MathWorks, Inc., Natick, MA) can also be used to implement the Hough transform.)

The SVM within Global SVM classifier 1242 is trained to classify the malignant and benign calcifications based on the selected features and the results of the local SVM analyzer 1222. A training data set of an approximately equal number of benign and cancer calcification cases are used to train the Global SVM analyzer 1242. The resulting SVM is tested on an independent test data set to evaluate its performance and generalization capability. The training process is iterated to select the optimal kernels and structures for the SVM. Using a multiple SVM configuration such as the example shown in FIG. 9, multiple SVMs may be provided to process the same training and test data sets, then selecting the SVM that provides the optimal output to process live data.

An enhanced version of a soft margin SVM is used in the preferred embodiment of the Global SVM classifier 1242. A traditional soft margin SVM is constructed by maximizing the functional

$$W(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i,j}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \tag{3}$$

subject to the constraints

$$\begin{gathered}\sum_{i=1}^{l}\alpha_i y_i=0\\ 0\le\alpha_i\le C,\quad i=1,2,\mathrm{K}\,,l\end{gathered} \tag{4}$$

The constant C is selected to penalize the misclassified points.

In the enhanced soft margin SVM, the constant C is not necessarily the same for all input vectors. In particular, one may choose different Cs for benign cases and malignant cases to associate different penalties with missed cancers and false alarms. The enhanced SVM is constructed by maximizing the functional

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j}^{l} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{5}$$

subject to the constraints

$$\begin{aligned} &\sum_{i=1}^{l} \alpha_i y_i = 0 \\ &0 \le \alpha_i \le C_i, \quad i = 1,2,\mathrm{K}\,,l \end{aligned} \tag{6}$$

Mass detection subsystem 1204 is similar to the calcification subsystem 1202. However, instead of calcification, the preprocessing steps of the subsystem 1204 are specifically designed to detect and segment masses and to extract features associated with the masses. The SVM training procedures are the same as the calcification subsystem 1202.

An important indicator of abnormalities is the asymmetric density patterns between the left and right images and the changes in mammogram images taken at different times. Detecting asymmetric dense regions can significantly improve the performance of the entire system. Clearly, it is not realistic to expect a perfect match even for symmetrical cases, therefore, the matching and registration algorithm used for asymmetry detection (step 1214) will allow normal small variations in the density patterns. The main focus of the algorithm will be the topological differences of the relatively high density areas between the two images. The procedure for asymmetry detection 1214 is as follows:

1. Construct two graphs representing the dense areas in the two images under comparison.

2. Find an optimal matching between the vertices of two graphs.

3. Evaluate the mismatched vertices and eliminate the ones that can be merged into adjacent vertices within acceptable variations.

4. The remaining mismatched vertices represent the asymmetric densities.

The appearances of masses in mammogram images are usually much more subtle than the calcifications. In mass segmentation step 1224, geometric transformation techniques are used to detect the often ill-defined boundaries. Hough transforms, described above, can be applied to detect specific shapes such as lines or circles in the images. Radon transforms are useful in handling irregular shapes.

Feature extraction step 1234 is performed in the same manner as the feature extraction step 1232 of calcification subsystem 1202. Important features to be extracted are location, size, shape, margins and x-ray attenuation. Evaluation of additional qualities, such as textures of the mass area, may also be useful for feature extraction in the mass detection subsystem 1204.

SVM classifier 1244 is trained and tested using a procedure similar to that used for Global SVM classifier 1242 in the calcification subsystem. SVM classifier 1244, comprising one or more SVMs, receives the output of feature extraction step 1234 and classifies the data into appropriate categories for each of the extracted features. For example, mass shape may have one of the following characteristics: round, oval, lobular or irregular, such that that SVM classifier 1244 would distribute the data into one of the four categories of shape characteristic. Similarly, there are five types of margins: circumscribed, obscured, micro-lobulated, ill-defined and spiculated, and SVM classifier would divide the data into one of the five margin categories. In view of the number of different mass-related features that are relevant to diagnosis of malignancy, it may be desirable to structure SVM classifier 1244 into a hierarchical configuration, assigning at least one first -level SVM to each feature, then combining the optimal outputs for processing through higher level SVMs until a single output is generated from SVM classifier 1244. This output is input to global SVM analyzer 1250 which combines the mass detection results with the results of the calcification and structure distortion subsystems to produce a diagnosis.

Structural distortion detection subsystem 1206 is similar to the calcification subsystem 1202. The preprocessing steps, spiculation detector 1216 and feature extraction 1226, are specifically designed to detect suspicious regions and extract features associated with structure distortions. Spiculations, which typically appear as radiating lines, or a “sunburst” pattern, can represent a desmoplastic process in conjunction with a possibly infiltrating tumor. On the other hand, postsurgical scarring from a previous

biopsy, radial scars, trauma, and infection may also produce a lesion with spiculated margins. The presence of spiculations in conjunction with the results of the other detection subsystems thus provide a good diagnostic tool. The SVM training procedures for SVM classifier 1236 are the same as for the classifiers previously described for the other detection subsystems. The output of SVM classifier 1236 will typically provide an output indicating the presence or not of spiculated distortions. This output is combined with the outputs of the other detection subsystems for input to overall SVM analyzer 1250 for use in the diagnosis of presence or not of a malignancy.

While the preceding example describes a procedure for analysis of mammograms for diagnosis of breast cancer, applications of computer-aided image analysis according to the present invention are not so limited, but are as wide-ranging as the applications of digital imaging itself. Generally, any situation in which a digital image is to be analyzed to aid in decision making, e.g., medical, industrial, geologic and space exploration, air or satellite reconnaissance, etc., or simply to provide information about the subject matter of the image where the image contains many data points that are subject to a number of interpretations, can benefit by employing image analysis according to present invention.

Alternative embodiments of the present invention will become apparent to those having ordinary skill in the art to which the present invention pertains. Such alternate embodiments are considered to be encompassed within the spirit and scope of the present invention. Accordingly, the scope of the present invention is described by the appended claims and is supported by the foregoing description.